MPLS Virtual Private Networks

Feature Overview

The IP virtual private network (VPN) feature for Multiprotocol Label Switching (MPLS) allows a Cisco IOS network to deploy scalable IPv4 Layer 3 VPN backbone services. An IP VPN is the foundation companies use for deploying or administering value-added services including applications and data hosting network commerce, and telephony services to business customers.

In private local area networks (LANs), IP-based intranets have fundamentally changed the way companies conduct their business. Companies are moving their business applications to their intranets to extend over a wide area network (WAN). Companies are also embracing the needs of their customers, suppliers, and partners by using extranets (an intranet that encompasses multiple businesses). With extranets, companies reduce business process costs by facilitating supply-chain automation, electronic data interchange (EDI), and other forms of network commerce. To take advantage of this business opportunity, service providers must have an IP VPN infrastructure that delivers private network services to businesses over a public infrastructure.

Tag Switching/MPLS Terminology

The following table lists the old Tag Switching terms and the new MPLS terms found in this document.

Old Designation	New Designation
Tag Switching	MPLS, Multiprotocol Label Switching
Tag (short for Tag Switching)	MPLS
Tag (item or packet)	Label
TDP (Tag Distribution Protocol)	LDP (Label Distribution Protocol) **Note** Cisco TDP and LDP (MPLS Label Distribution Protocol)) are nearly identical in function, but use incompatible message formats and some different procedures. Cisco will be changing from TDP to a fully compliant LDP.
Tag Switched	Label Switched
TFIB (Tag Forwarding Information Base)	LFIB (Label Forwarding Information Base)
TSR (Tag Switching Router)	LSR (Label Switching Router)

Old Designation	New Designation
TSC (Tag Switch Controller)	LSC (Label Switch Controller)
ATM-TSR	ATM-LSR (ATM Label Switch Router, for example, BPX 8650.)
TVC (Tag VC, Tag Virtual Circuit)	LVC (Label VC, Label Virtual Circuit)
TSP (Tag Switch Path)	LSP (Label Switch Path)
XTagATM (extended Tag ATM port)	XmplsATM (extended MPLS ATM port)

IP Virtual Private Networks

To effectively implement an IP VPN in your facility, ensure your IP VPN meets the following basic requirements:

Privacy—All IP VPNs offer privacy over a shared (public) network infrastructure. Most companies use an encrypted tunnel. This is only one of several ways to provide network and data privacy.

Scalability—For proper service delivery, VPNs must scale to serve hundreds of thousands of sites and users. Besides being a managed service, VPNs are also a management tool for service providers to control access to services. One example is Closed User Groups for data and voice services.

Flexibility—IP VPNs must handle the any-to-any traffic patterns characteristic of corporate intranets and extranets, in which data no longer flows to and from a central location. VPNs must also have the inherent flexibility to add new sites quickly, connect users over different media, and meet the increasingly sophisticated transport and bandwidth requirements of new intranet applications.

Predictable Performance—Performance needs vary widely requiring different classes of service, but the common requirement is that the performance is predictable. Examples of the ranges of performance requirements include:

- Remote access for mobile users—Require widespread connectivity
- Branch offices—Require a sustained performance level because of the interactive nature of the intranet application in a branch office
- Video conferencing—Require specific performance characteristics

MPLS Virtual Private Networks

MPLS VPNs allow service providers to deploy scalable VPNs and build the foundation to deliver value-added services, including:

Connectionless Service—A significant technical advantage of MPLS VPNs is that they are connectionless. The Internet owes its success to its basic technology, TCP/IP. TCP/IP is built on packet-based, connectionless network paradigm. This means that no prior action is necessary to establish communication between hosts, making it easy for two parties to communicate. To establish privacy in a connectionless IP environment, current VPN solutions impose a connection-oriented, point-to-point overlay on the network. Even if it runs over a connectionless network, a VPN cannot take advantage of the ease of connectivity and multiple services available in connectionless networks. When you create a connectionless VPN, you do not need tunnels and encryption for network privacy, thus eliminating significant complexity.

Centralized Service—Building VPNs in Layer 3 allows delivery of targeted services to a group of users represented by a VPN. A VPN must give service providers more than a mechanism for privately connecting users to intranet services. It must also provide a way to flexibly deliver value-added services to targeted customers. Scalability is critical, because customers want to use services privately in their intranets and extranets. Because MPLS VPNs are seen as private intranets, you may use new IP services such as:

- multicast
- quality of service (QoS)
- telephony support within a VPN
- centralized services including content and web hosting to a VPN

You can customize several combinations of specialized services for individual customers. For example, a service that combines IP multicast with a low-latency service class enables videoconferencing within an intranet.

Scalability—If you create a VPN using connection-oriented, point-to-point overlays, Frame Relay, or ATM virtual connections (VCs), the VPN's key deficiency is scalability. Specifically, connection-oriented VPNs without fully meshed connections between customer sites, are not optimal. MPLS-based VPNs instead use the peer model and Layer 3 connectionless architecture to leverage a highly scalable VPN solution. The peer model requires a customer site to only a peer with one provider edge (PE) router as opposed to all other CPE or customer edge (CE) routers that are members of the VPN. The connectionless architecture allows the creation of VPNs in Layer 3, eliminating the need for tunnels or VCs.

Other scalability issues of MPLS VPNs are due to the partitioning of VPN routes between PE routers and the further partitioning of VPN and IGP routes between PE routers and provider (P) routers in a core network.

- PE routers must maintain VPN routes for those VPNs who are members.
- P routers do not maintain any VPN routes.

This increases the scalability of the provider's core and ensures that no one device is a scalability bottleneck.

Security—MPLS VPNs offer the same level of security as connection-oriented VPNs. Packets from one VPN do not inadvertently go to another VPN. Security is provided

1. At the edge of a provider network, ensuring packets received from a customer are placed on the correct VPN.
2. At the backbone, VPN traffic is kept separate. Malicious spoofing (an attempt to gain access to a PE router) is nearly impossible because the packets received from customers are IP packets. These IP packets must be received on a particular interface or subinterface to be uniquely identified with a VPN label.

Easy to Create—To take full advantage of VPNs, it must be easy for customers to create new VPNs and user communities. Because MPLS VPNs are connectionless, no specific point-to-point connection maps or topologies are required. You can add sites to intranets and extranets and form closed user groups. When you manage VPNs in this manner, it enables membership of any given site in multiple VPNs, maximizing flexibility in building intranets and extranets.

Flexible Addressing—To make a VPN service more accessible, customers of a service provider can design their own addressing plan, independent of addressing plans for other service provider customers. Many customers use private address spaces, as defined in RFC 1918, and do not want to invest the time and expense of converting to public IP addresses to enable intranet connectivity. MPLS VPNs allow customers to continue to use their present address spaces without network

address translation (NAT) by providing a public and private view of the address. A NAT is required only if two VPNs with overlapping address spaces want to communicate. This enables customers to use their own unregistered private addresses, and communicate freely across a public IP network.

Integrated Class of Service (CoS) Support—CoS is an important requirement for many IP VPN customers. It provides the ability to address two fundamental VPN requirements:

1. Predictable performance and policy implementation
2. Support for multiple levels of service in a MPLS VPN

Network traffic is classified and labeled at the edge of the network before traffic is aggregated according to policies defined by subscribers and implemented by the provider and transported across the provider core. Traffic at the edge and core of the network can then be differentiated into different classes by drop probability or delay.

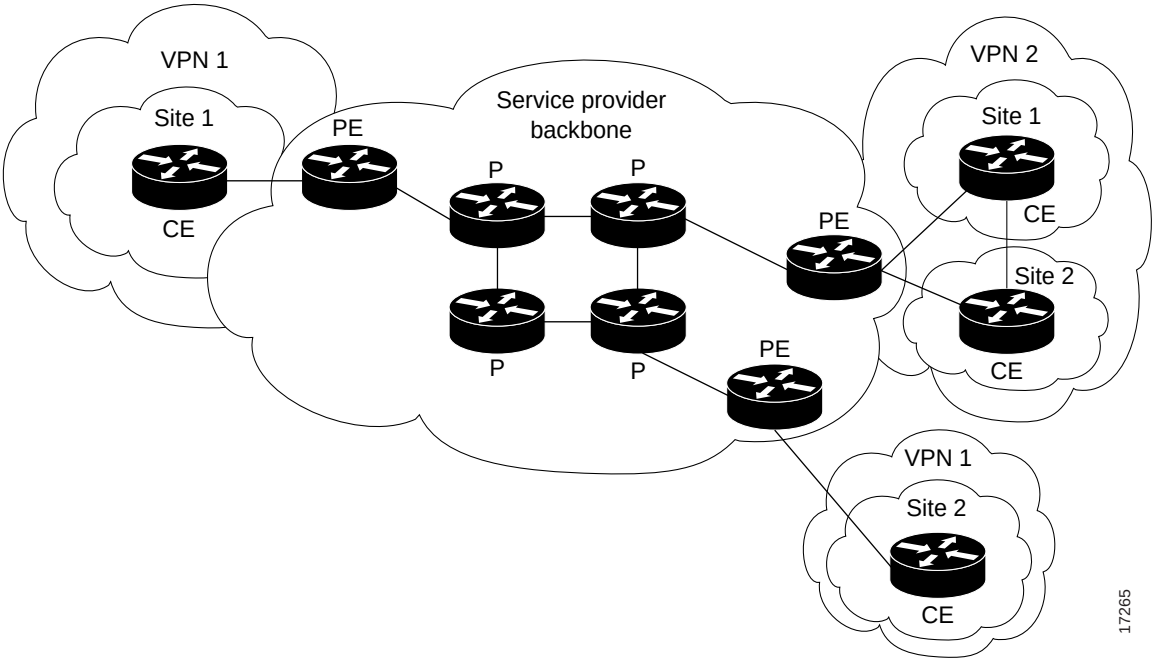
Straightforward Migration—For service providers to quickly deploy VPN services, use a straightforward migration path. MPLS VPNs are unique because you can be build them over multiple network architectures, including IP, ATM, Frame Relay, and hybrid networks.

Migration for the end customer is simplified because there is no requirement to support MPLS on the customer edge (CE) router and no modifications are required to a customer's intranet.

For a list of platforms supported by MPLS VPNs, refer to the section entitled Supported Platforms.

Figure 1 shows an example of a VPN with a service provider (P) backbone network, service provider edge routers (PE), and customer edge routers (CE).

Figure 1 VPNs with a Service Provider Backbone

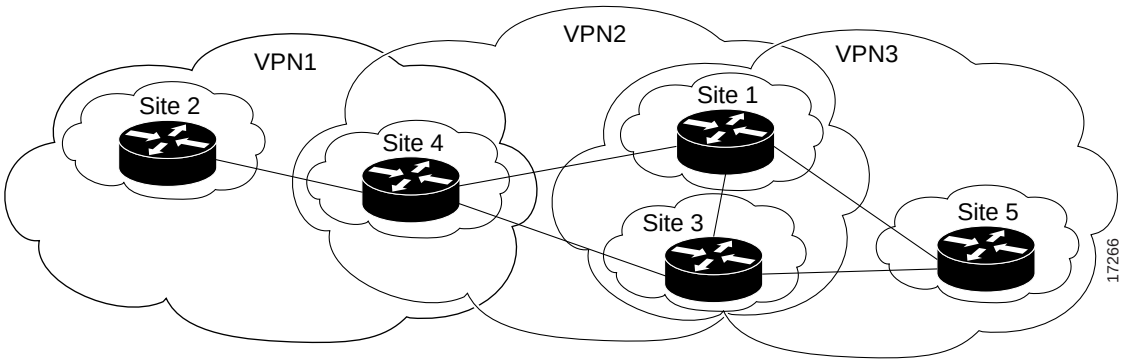


A VPN contains customer devices attached to the CE routers. These customer devices use VPNs to exchange information between devices. Only the PE routers are aware of the VPNs.

Figure 2 shows five customer sites communicating within three VPNs. The VPNs can communicate with the following sites:

- VPN1—sites 2 and 4
- VPN2—sites 1, 3, and 4
- VPN3—sites 1,3, and 5

Figure 2 Customer Sites within VPNs



VPN Operation

Each VPN is associated with one or more VPN routing/forwarding instances (VRFs). A VRF defines the VPN membership of a customer site attached to a PE router. A VRF consists of an IP routing table, a derived Cisco Express Forwarding (CEF) table, a set of interfaces that use the forwarding table, and a set of rules and routing protocol parameters that control the information that is included into the routing table.

A one-to-one relationship does not necessarily exist between customer sites and VPNs. A given site can be a member of multiple VPNs, as shown in Figure 2. However, a site can only associate with one (and only one) VRF. A customer site's VRF contains all the routes available to the site from the VPNs of which it is a member.

Packet forwarding information is stored in the IP routing table and the CEF table for each VRF. A separate set of routing and CEF tables is maintained for each VRF. These tables prevent information from being forwarded outside a VPN, and also prevent packets that are outside a VPN from being forwarded to a router within the VPN.

VPN Route Target Communities

The distribution of VPN routing information is controlled through the use of VPN route target communities, implemented by border gateway protocol (BGP) extended communities. Distribution of VPN routing information works as follows:

- When a VPN route learned from a CE router is injected into BGP, a list of VPN route target extended community attributes are associated with it. Typically the list of route target community values is set from an export list of route targets associated with the VRF from which the route was learned.

- An import list of route target extended communities is associated with each VRF. The import list defines route target extended community attributes a route must have for the route to be imported into the VRF. For example, if the import list for a particular VRF includes route target communities A, B, and C, then any VPN route that carries any of those route target extended communities — A, B, *or* C — is imported into the VRF.

BGP Distribution of VPN Routing Information

A service provider edge (PE) router can learn an IP prefix from a customer edge (CE) router by static configuration, through a BGP session with the CE router, or through the routing information protocol (RIP) exchange with the CE router. The IP prefix is a member of the IPv4 address family. After it learns the IP prefix, the PE converts it into a VPN-IPv4 prefix by combining it with an 8-byte route distinguisher (RD). The generated prefix is a member of the VPN-IPv4 address family. It serves to uniquely identify the customer address, even if the customer site is using globally nonunique (unregistered private) IP addresses.

The route distinguisher used to generate the VPN-IPv4 prefix is specified by a configuration command associated with the VRF on the PE router.

BGP distributes reachability information for VPN-IPv4 prefixes for each VPN. BGP communication takes place at two levels: within IP domains, known as an autonomous systems (interior BGP or IBGP) and between autonomous systems (external BGP or EBGP). PE-PE or PE-RR (route reflector) sessions are IBGP sessions, and PE-CE sessions are EBGP sessions.

BGP propagates reachability information for VPN-IPv4 prefixes among PE routers by means of the BGP multiprotocol extensions (see RFC 2283, *Multiprotocol Extensions for BGP-4)* which define support for address families other than IPv4. It does this in a way that ensures the routes for a given VPN are learned only by other members of that VPN, enabling members of the VPN to communicate with each other.

MPLS Forwarding

Based on routing information stored in the VRF IP routing table and VRF CEF table, packets are forwarded to their destination using MPLS.

A PE router binds a label to each customer prefix learned from a CE router and includes the label in the network reachability information for the prefix that it advertises to other PE routers. When a PE router forwards a packet received from a CE router across the provider network it labels the packet with the label learned from the destination PE router. When the destination PE router receives the labeled packet it pops the label and uses it to direct the packet to the correct CE router. Label forwarding across the provider backbone, is based on either dynamic label switching or traffic engineered paths. A customer data packet carries two levels of labels when traversing the backbone:

1. Top label directs the packet to the correct PE router
2. Second label indicates how that PE router should forward the packet to the CE router

Benefits

This section describes the benefits of VPNs in general and MPLS VPNs in particular.

IP VPNs are attractive because they:

1. Reduce the cost of connecting branch offices, telecommuters, and mobile users to a corporate intranet, which operate over the public infrastructure of the Internet
2. Are more cost-effective than private WANs constructed with leased lines

However, conventional VPNs do not scale well. They are based on creating and maintaining a full mesh of tunnels or permanent virtual circuits among all sites belonging to a particular VPN, using:

- IPSec
- Layer 2 tunneling protocol (L2TP)
- Layer 2 forwarding (L2F) protocol
- generic routing encapsulation (GRE)
- Frame Relay
- ATM protocols

The overhead required to provision and manage these connection-based schemes cannot be supported in a provider network that must support hundreds or thousands of VPNs, each with tens or hundreds or thousands of sites and thousands or tens of thousands of routes.

MPLS VPNs, which are created in Layer 3, are connectionless, and therefore substantially more scalable and easier to build and manage than conventional VPNs. In addition, you can add value-added services, such as application and data hosting, network commerce, and telephony services to a particular MPLS VPN because the service provider's backbone recognizes each MPLS VPN as a separate, connectionless IP network.

MPLS VPNs offer:

- A platform for rapid deployment of additional value-added IP services, including intranets, extranets, voice, multimedia, and network commerce
- Privacy and security equal to that provided by Layer-2 VPNs by limiting the distribution of a VPN's routes to only those routers that are members of the VPN
- Seamless integration with customer intranets
- Increased scalability over current VPN implementations, with thousands of sites per VPN and hundreds of thousands of VPNs per service provider
- IP Class of Service (CoS), with support for multiple classes of service and priorities within VPNs, as well as between VPNs
- Management of VPN membership and provisioning of new VPNs for rapid deployment
- Scalable any-to-any connectivity for extended intranets and extranets that encompass multiple businesses

Related Features and Technologies

VPNs may be used with the Class of Service (CoS) feature for MPLS.

Related Documents

- *MPLS Class of Service Feature Guide*
- *Cisco IOS Release 12.0 Network Protocols Command Reference, Part I*
- Internet draft *draft-rosen-vpn-mpls-00.txt* VPN architecture description
- RFC 1163, *A Border Gateway Protocol*
- RFC 1164, *Application of the Border Gateway Protocol in the Internet*
- RFC 2283, *Multiprotocol Extensions for BGP-4*
- RFC 2547, *BGP/MPLS VPNs*
- Internet draft *draft-rekhter-bgp-mpls-00.txt*, Carrying Label information in BGP-4
- Internet draft *draft-ramachandra-bgp-ext-communities-01.txt* extended community attributes

Supported Platforms

The following is a list of router platforms supported at the provider core.

- Cisco 7200 series
- Cisco 7500 series
- Cisco 8540 series (MSR)
- Cisco 8650 series (BPX)
- Cisco 8800 series (MGX)

The following is a list of router platforms supported at the provider edge.

- Cisco 3640 series
- Cisco 7200 series
- Cisco 7500 series

Supported Standards, MIBs and RFCs

MIBs

No new or modified MIBs are supported by this feature.

RFCs

- RFC 1163, *A Border Gateway Protocol*
- RFC 1164, *Application of the Border Gateway Protocol in the Internet*
- RFC 2283, *Multiprotocol Extensions for BGP-4*
- RFC 2547, *BGP/MPLS VPNs*

Standards

No new or modified standards are supported by this feature.

Prerequisites

Your network must be running the following Cisco IOS services before you configure VPN operation:

- MPLS in provider backbone routers, or GRE tunnel connectivity among all provider edge (PE) routers
- MPLS with VPN code in provider routers with VPN edge service (PE) routers
- BGP in all routers providing a VPN service
- CEF switching in every MPLS-enabled router
- CoS feature (optional)

Configuration Tasks

Perform the following tasks to configure and verify VPNs:

- Defining VPNs
- Configuring BGP PE to PE Routing Sessions
- Configuring BGP PE to CE Routing Sessions
- Configuring RIP PE to CE Routing Sessions
- Configuring Static Route PE to CE Routing Sessions
- Verifying VPN Operation

Defining VPNs

To define VPN routing instances, perform the following steps on the PE router:

Step	Command	Purpose
1	`Router(config)# ip vrf vrf-name`	Enter VRF configuration mode and define the VPN routing instance by assigning a VRF name.
2	`Router(config-vrf)# rd route-distinguisher`	Create routing and forwarding tables.
3	`Router(config-vrf)# route-target {import | export | both} route-target-ext-community`	Create a list of import and/or export route target communities for the specified VRF.
4	`Router(config-vrf)# import map route-map`	(Optional) Associate the specified route map with the VRF.
5	`Router(config-if)# ip vrf forwarding vrf-name`	Associate a VRF with an interface or subinterface.

Configuring BGP PE to PE Routing Sessions

To configure BGP PE to PE routing sessions in a provider network, perform the following steps on the PE routers:

Step	Command	Purpose
1	`Router(config)# router bgp autonomous-system`	Configures the IBGP routing process with the autonomous system number passed along to other IBGP routers.
2	`Router(config-router)# neighbor {ip-address | peer-group-name} remote-as number`	Specifies a neighbor's IP address or IBGP peer group identifying it to the local autonomous system.
3	`Router(config-router)# neighbor ip-address activate`	Activates the advertisement of the IPv4 address family.

Configuring BGP PE to CE Routing Sessions

To configure BGP PE to CE routing sessions perform the following steps on the PE router:

Step	Command	Purpose
1	`Router(config)# router bgp autonomous-system`	Configures a EBGP routing process with the autonomous system number passed along to other EBGP routers.
2	`Router(config-router)# neighbor {ip-address | peer-group-name} remote-as number`	Specifies a neighbor's IP address or EBGP peer group identifying it to the local autonomous system.
3	`Router(config-router)# neighbor ip-address activate`	Activates the advertisement of the IPv4 address family.

Configuring RIP PE to CE Routing Sessions

To configure RIP PE to CE routing sessions perform the following steps on the PE router:

Step	Command	Purpose
1	`Router(config)# router rip`	Enables RIP.
2	`Router(config-router)# address-family ipv4 [unicast] vrf vrf-name`	Defines RIP parameters for PE to CE routing sessions. **Note** The default is Off for auto-summary and synchronization in the VRF address-family submode.
3	`Router(config-router)# network prefix`	Enables RIP on the PE to CE link.

Configuring Static Route PE to CE Routing Sessions

To configure static route PE to CE routing sessions perform the following steps on the PE router:

Step	Command	Purpose
1	`Router(config)# ip route vrf vrf-name`	Defines static route parameters for every PE to CE session.
2	`Router(config-router)# address-family ipv4 [unicast] vrf vrf-name`	Defines static route parameters for every BGP PE to CE routing session. **Note** The default is Off for auto-summary and synchronization in the VRF address-family submode.
3	`Router(config-router)# redistribute static`	Redistributes VRF static routes into the VRF BGP table.
4	`Router(config-router)# redistribute static connected`	Redistributes directly connected networks into the VRF BGP table.

Verifying VPN Operation

To verify VPN operation, perform the following steps:

Step	Command	Purpose
1	Router# **show ip vrf**	Displays the set of defined VRFs and interfaces.
2	Router# **show ip vrf** [{**brief** | **detail** | **interfaces**}] *vrf-name*	Displays information about defined VRFs and associated interfaces.
3	Router# **show ip route vrf** *vrf-name*	Displays the IP routing table for a VRF.
4	Router# **show ip protocols vrf** *vrf-name*	Displays the routing protocol information for a VRF.
5	Router# **show ip cef vrf** *vrf-name*	Displays the CEF forwarding table associated with a VRF.
6	Router# **show ip interface** *interface-number*	Displays the VRF table associated with an interface.
7	Router# **show ip bgp vpnv4 all** [**tags**]	Displays information about all BGPs.
8	Router# **show tag-switching forwarding vrf** *vrf-name* [*prefix mask/length*][**detail**]	Displays label forwarding entries that correspond to VRF routes advertised by this router.

Configuration Examples

This section provides a sample configuration file from a PE router.

```
ip cef distributed           ! CEF switching is pre-requisite for label Switching
frame-relay switching
!
ip vrf vrf1                  ! Define VPN Routing instance vrf1
 rd 100:1
 route-target both 100:1     ! Configure import and export route-targets for vrf1
!
ip vrf vrf2                  ! Define VPN Routing instance vrf2
 rd 100:2
 route-target both 100:2     ! Configure import and export route-targets for vrf2
 route-target import 100:1   ! Configure an additional import route-target for vrf2
 import map vrf2_import      ! Configure import route-map for vrf2
!
interface lo0
 ip address 10.13.0.13 255.255.255.255
!
interface atm9/0/0           ! Backbone link to another Provider router
!
interface atm9/0/0.1 tag-switching
 ip unnumbered loopback0
 no ip directed-broadcast
 tag-switching atm vpi 2-5
 tag-switching ip

interface atm5/0
 no ip address
 no ip directed-broadcast
 atm clock INTERNAL
 no atm ilmi-keepalive

interface Ethernet1/0
 ip address 3.3.3.5 255.255.0.0
 no ip directed-broadcast
 no ip mroute-cache
 no keepalive

interface Ethernet5/0/1      ! Set up Ethernet interface as VRF link to a CE router
 ip vrf forwarding vrf1
 ip address 10.20.0.13 255.255.255.0
 !
interface hssi 10/1/0

 hssi internal-clock
 encaps fr
 frame-relay intf-type dce
 frame-relay lmi-type ansi
!
interface hssi 10/1/0.16 point-to-point
 ip vrf forwarding vrf2
 ip address 10.20.1.13 255.255.255.0
 frame-relay interface-dlci 16 ! Set up Frame Relay PVC subinterface as link to another
!                             ! CE router

router bgp 1                 ! Configure BGP sessions
 no synchronization
 no bgp default ipv4-activate              ! Deactivate default IPv4 advertisements
 neighbor 10.15.0.15 remote-as 1           ! Define IBGP session with another PE
 neighbor 10.15.0.15 update-source lo0
!
 address-family vpnv4 unicast              ! Activate PE exchange of VPNv4 NLRI
```

```
  neighbor 10.15.0.15 activate
  exit-address-family
!
 address-family ipv4 unicast vrf vrf1       ! Define BGP PE-CE session for vrf1
  redistribute static
  redistribute connected
  neighbor 10.20.0.60 remote-as 65535
  neighbor 10.20.0.60 activate
  no auto-summary
  exit-address-family
!
 address-family ipv4 unicast vrf vrf2       ! Define BGP PE-CE session for vrf2
  redistribute static
  redistribute connected
  neighbor 10.20.1.11 remote-as 65535
  neighbor 10.20.1.11 update-source h10/1/0.16
  neighbor 10.20.1.11 activate
  no auto-summary
  exit-address-family
!
! Define a VRF static route
ip route vrf vrf1 12.0.0.0 255.0.0.0 e5/0/1 10.20.0.60
!
route-map vrf2_import permit 10             ! Define import route-map for vrf2.
 ...
```

Command Reference

This section documents new or modified commands. All other commands used with this feature are documented in the Cisco IOS Release 12.0 command references.

- address-family
- clear ip route vrf
- exit-address-family
- import map
- ip route vrf
- ip vrf forwarding
- ip vrf
- neighbor activate
- rd
- route-target
- show ip bgp vpnv4
- show ip cef vrf
- show ip protocols vrf
- show ip route vrf
- show ip vrf
- show tag-switching forwarding vrf

In Cisco IOS Release 12.0(1)T or later, you can search and filter the output for **show** and **more** commands. This functionality is useful when you need to sort through large amounts of output, or if you want to exclude output that you do not need to see.

To use this functionality, enter a **show** or **more** command followed by the "pipe" character (|), one of the keywords **begin**, **include**, or **exclude**, and an expression that you want to search or filter on:

command **|** {**begin** | **include** | **exclude**} *regular-expression*

Following is an example of the **show atm vc** command in which you want the command output to begin with the first line where the expression "PeakRate" appears:

show atm vc | begin PeakRate

For more information on the search and filter functionality, refer to the Cisco IOS Release 12.0(1)T feature module titled *CLI String Search*.

Command Conventions

boldface font	Commands and keywords are in **boldface**.
italic font	Arguments for which you supply values are in *italics*. In contexts that do not allow *italics*, arguments are enclosed in **angle brackets** < >.
[ ]	Elements in square **brackets** are optional.
{ **x** | **y** | **z** }	Required alternative keywords are grouped in **braces** and separated by vertical bars.
[**x** | **y** | **z**]	Optional alternative keywords are grouped in **brackets** and separated by vertical bars.

address-family

To enter the address family submode for configuring routing protocols, such as BGP, RIP and static routing, use the **address-family** global configuration command. To disable the address family submode for configuring routing protocols, use the **no** form of this command.

VPN-IPv4 unicast

address-family vpnv4 [**unicast**]

no address-family vpnv4 [**unicast**]

IPv4 unicast

address-family ipv4 [**unicast**]

no address-family ipv4 [**unicast**]

IPv4 unicast with CE router

address-family ipv4 [**unicast**] **vrf** *vrf-name*

no address-family ipv4 [**unicast**] **vrf** *vrf-name*

Syntax Description

ipv4	Configures sessions that carry standard IPv4 address prefixes.
vpnv4	Configures sessions that carry customer VPN-IPv4 prefixes, each of which has been made globally unique by adding an 8-byte route distinguisher.
unicast	(Optional) Specifies unicast prefixes.
vrf *vrf-name*	Specifies the name of a VPN routing/forwarding instance (VRF) to associate with submode commands.

Default

Routing information for address family IPv4 is advertised by default when you configure a BGP session using the **neighbor**...**remote-as** command unless you execute the **no bgp default ipv4-activate** command.

Command Mode

Router configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Using the **address-family** command puts you in address family configuration submode (prompt: `(config-router-af)#`). Within this submode, you can configure address-family specific parameters for routing protocols, such as BGP, that can accommodate multiple Layer 3 address families.

To leave address family configuration submode and return to router configuration mode, type **exit-address-family**, or simply **exit**.

Examples

The address-family command in the following example puts the router into address family configuration submode for the VPNv4 address family. Within the submode, you can configure advertisement of NLRI for the VPNv4 address family using **neighbor activate** and other related commands:

```
(config)# router bgp 100
(config-router)# address-family vpnv4
(config-router-af)#
```

The command in the following example puts the router into address family configuration submode for the IPv4 address family. Use this form of the command, which specifies a VRF, only to configure routing exchanges between PE and CE devices. This address-family command causes subsequent commands entered in the submode to be executed in the context of VRF vrf2. Within the submode, you can use **neighbor activate** and other related commands to accomplish the following:

- Configure advertisement of IPv4 NLRI between the PE and CE routers.
- Configure translation of the IPv4 NLRI (that is, translate IPv4 into VPNv4 for NLRI received from the CE, and translate VPNv4 into IPv4 for NLRI to be sent from the PE to the CE).
- Enter the routing parameters that apply to this VRF.

Entered the address family submode as follows:

```
(config)# router bgp 100
(config-router)# address-family ipv4 unicast vrf vrf2
(config-router-af)#
```

Related Commands

Command	Description
exit-address-family	Exits address family submode.
neighbor activate	Exchanges an address with a neighboring router.

clear ip route vrf

To remove routes from the VRF routing table, use the **clear ip route vrf** EXEC command.

clear ip route vrf *vrf-name* {***** | ***network*** [***mask***]}

Syntax Description

vrf-name	Name of the VPN routing/forwarding instance (VRF) for the static route.
*	Deletes all routes for a given VRF.
network	Destination to be removed, in dotted-decimal format.
mask	(Optional) Mask for the specified network destination, in dotted-decimal format.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced

Usage Guidelines

Use this command to clear routes from the routing table. Use the asterisk (*) to delete all routes from the forwarding table for a specified VRF, or enter the address and mask of a particular network to delete the route to that network.

Example

The following command removes the route to the network 10.13.0.0 in the vpn1 routing table:

```
Router# clear ip route vrf vpn1 10.13.0.0
```

Related Command

Command	Description
show ip route vrf	Displays the IP routing table associated with a VRF.

exit-address-family

To exit from the address family submode, use the **exit-address-family** address family submode command.

exit-address-family

Syntax Description

This command has no arguments or keywords.

Default

No default behavior or values.

Command Mode

Address family submode

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

This command can be abbreviated to **exit**.

Example

The following example shows how to exit the address-family command mode:

```
(config-router-af)# exit-address-family
```

Related Commands

Command	Description
address-family	Enters the address family submode used to configure routing protocols.

import map

To configure an import route map for a VRF, use the **import** VRF submode command.

import map *route-map*

Syntax Description

route-map	Specifies the route map to be used as an import route map for the VRF.

Default

There is no default. A VRF has no import route map unless one is configured using the **import map** command.

Command Mode

VRF submode

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Use an import route map when an application requires finer control over the routes imported into a VRF than provided by the import and export extended communities configured for the importing and exporting VRF.

The **import-map** command associates a route map with the specified VRF. You can filter routes that are eligible for import into a VRF, based on the route target extended community attributes of the route, through the use of a route map. The route map might deny access to selected routes from a community that is on the import list.

Example

The following example shows how to configure an import route map for a VRF:

```
(config)# ip vrf vrf_blue
(config-vrf)# import map blue_import_map
```

Related Commands

Command	Description
ip vrf	Enters VRF configuration mode.
route-target	Configures import and export extended community attributes for the VRF.

show ip vrf	Displays information about a VRF or all VRFs.

ip route vrf

To establish static routes for a VRF, use the **ip route vrf** global configuration command. To disable static routes, use the **no** form of this command.

ip route vrf *vrf-name prefix mask* [*next-hop-address*] [*interface* {*interface-number*}] [**global**] [*distance*] [**permanent**] [**tag** *tag*]

no ip route vrf *vrf-name prefix mask* [*next-hop-address*] [*interface* {*interface-number*}] [**global**] [*distance*] [**permanent**] [**tag** *tag*]

Syntax Description

vrf-name	Name of the VPN routing/forwarding instance (VRF) for the static route.
prefix	IP route prefix for the destination, in dotted-decimal format.
mask	Prefix mask for the destination, in dotted-decimal format.
next-hop-address	(Optional) IP address of the next hop (the forwarding router that can be used to reach that network).
interface	(Optional) Type of network interface to use: ATM, Ethernet, loopback, POS (packet over SONET), or null.
interface-number	Number identifying the network interface to use.
global	Specifies that the given next hop address is in the non-VRF routing table.
distance	(Optional) An administrative distance for this route.
permanent	(Optional) Specifies that this route will not be removed, even if the interface shuts down.
tag *tag*	(Optional) Label value that can be used for controlling redistribution of routes through route maps.

Default

No default behavior or values.

Command Mode

Global configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Use a static route when the Cisco IOS software cannot dynamically build a route to the destination.

If you specify an administrative distance when you set up a route, you are flagging a static route that can be overridden by dynamic information. For example, IGRP-derived routes have a default administrative distance of 100. To set a static route to be overridden by an IGRP dynamic route, specify an administrative distance greater than 100. Static routes each have a default administrative distance of 1.

Static routes that point to an interface are advertised through RIP, IGRP, and other dynamic routing protocols, regardless of whether the routes are redistributed into those routing protocols. That is, static routes configured by specifying an interface lose their static nature when installed into the routing table.

However, if you define a static route to an interface not defined in a network command, no dynamic routing protocols advertise the route unless a redistribute static command is specified for these protocols.

Example

The following command reroutes packets addressed to network 137.23.0.0 in VRF vpn3 to router 131.108.6.6:

```
(config)# ip route vrf vpn3 137.23.0.0 255.255.0.0 131.108.6.6
```

Related Command

Command	Description
show ip route vrf	Displays the IP routing table associated with a VRF.

ip vrf forwarding

To associate a VRF with an interface or subinterface, use the **ip vrf forwarding** interface configuration command. To disassociate a VRF, use the **no** form of this command.

ip vrf forwarding *vrf-name*

no ip vrf forwarding *vrf-name*

Syntax Description

vrf-name	Name assigned to a VRF.

Default

The default for an interface is the global routing table.

Command Modes

Global configuration

Interface configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Use this command to associate an interface with a VRF. Executing this command on an interface removes the IP address. The IP address should be reconfigured.

Example

The following example shows how to link a VRF to ATM interface 0/0:

```
(config)# interface atm0/0
(config-if)# ip vrf forwarding vpn1
```

Related Commands

Command	Description
ip vrf	Defines a VRF.
ip route vrf	Establishes static routes for a VRF.

ip vrf

To configure a VRF routing table, use the **ip vrf** global configuration command. To remove a VRF routing table, use the **no** form of this command.

ip vrf *vrf-name*
no ip vrf *vrf-name*

Syntax Description

vrf-name	Name assigned to a VRF.

Defaults

No VRFs are defined. No import or export lists are associated with a VRF. No route maps are associated with a VRF.

Command Modes

Router configuration

Global configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The **ip vrf** *vrf-name* command creates a VRF routing table and a CEF (forwarding) table, both named *vrf-name*. Associated with these tables is the default route distinguisher value *route-distinguisher.*

Example

The following example imports a route map to a VRF:

```
(Router-config)# ip vrf vpn1
(config-vrf)# rd 100:2
 route-target both 100:2
 route-target import 100:1
```

Related Command

Command	Description
ip vrf forwarding	Associates a VRF with an interface or subinterface.

neighbor activate

To enable the exchange of information with a BGP neighboring router, use the **neighbor activate** router configuration command. To disable the exchange of an address with a neighboring router, use the **no** form of this command.

neighbor {*ip-address* | *peer-group-name*} **activate**

no neighbor {*ip-address* | *peer-group-name*} **activate**

Syntax Description

ip-address	IP address of the neighboring router.
peer-group-name	Name of BGP peer group.

Defaults

The exchange of addresses with neighbors is enabled by default for theVPN IPv4 address family. You can disable IPv4 address exchange using the general command **no default bgp ipv4 activate**, or you can disable it for a particular neighbor using the **no** form of this command.

For all other address families, address exchange is disabled by default. You can explicitly activate the default command using the appropriate address family submode.

Command Modes

Router configuration

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Use this command to enable or disable the exchange of addresses with a neighboring router.

Example

In the following example, a BGP router activates the exchange of a customer's IP address 10.15.0.15 to a neighboring router.

```
router bgp 100
 neighbor 10.15.0.15 remote-as 100
 neighbor 10.15.0.15 update-source loopback0
 address-family vpnv4 unicast
 neighbor 10.15.0.15 activate
 exit-address-family
```

Related Commands

Command	Description
address-family	Enters the address family submode.
exit-address-family	Exits the address family submode.

rd

To create routing and forwarding tables for a VRF, use the **rd** VRF submode command.

rd *route-distinguisher*

Syntax Description

route-distinguisher	Adds an 8-byte value to an IPv4 prefix to create a VPN IPv4 prefix.

Default

There is no default. An RD must be configured for a VRF to be functional.

Command Mode

VRF submode

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

A route distinguisher (RD) creates routing and forwarding tables and specifies the default route-distinguisher for a VPN. The RD is added to the beginning of the customer's IPv4 prefixes to change them into globally unique VPN-IPv4 prefixes.

An RD is either ASN-relative, in which case it is composed of an autonomous system number and an arbitrary number, or it is IP-address-relative, in which case it is composed of an IP address and an arbitrary number.

You can enter an RD in either of these formats:

16-bit AS number: *your 32-bit number*
For example, 101:3

32-bit IP address: *your 16-bit number*
For example, 192.168.122.15:1

Example

The following example configures a default RD for two VRFs. It illustrates the use of both AS-relative and IP address-relative RDs:

```
(config)# ip vrf vrf_blue
(config-vrf)# rd 100:3
(config-vrf)# ip vrf vrf_red
(config-vrf)# rd 173.13.0.12:200
```

Related Commands

Command	Description
ip vrf	Enters VRF configuration mode.
show ip vrf	Displays information about a VRF.

route-target

To create a route-target extended community for a VRF, use the **route-target** VRF submode command. To disable the configuration of a route-target community option, use the **no** form of this command.

route-target {**import | export | both**} *route-target-ext-community*

no route-target {**import | export | both**} *route-target-ext-community*

Syntax Description

import	Imports routing information from the target VPN extended community.
export	Exports routing information to the target VPN extended community.
both	Imports both import and export routing information to the target VPN extended community.
route-target-ext-community	Adds the route-target extended community attributes to the VRF's list of import, export, or both (import and export) route-target extended communities.

Default

There are not defaults. A VRF has no route-target extended community attributes associated with it until specified by the **route-target** command.

Command Mode

VRF submode

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

The **route-target** command creates lists of import and export route target extended communities for the specified VRF. Execute the command one time for each target community. Learned routes that carry a specific route target extended community are imported into all VRFs configured with that extended community as an import route target. Routes learned from a VRF site (for example, by BGP, RIP, or static route configuration) contain export route targets for extended communities configured for the VRF added as route attributes to control the VRFs into which the route is imported.

The route-target specifies a target VPN extended community. Like a route-distinguisher, an extended community is composed of either an autonomous system number and an arbitrary number, or an IP address and an arbitrary number. You can enter the numbers in either of these formats:

- *16-bit AS number*: *your 32-bit number*
  For example, 101:3
- *32-bit IP address*: *your 16-bit number*
  For example, 192.168.122.15:1

Example

The following example shows how to configure route-target extended community attributes for a VRF. The result of the command sequence is that VRF *vrf_blue* has two export extended communities (1000:1 and 1000:2) and two import extended communities (1000:1 and 173.27.0.130:200).

```
(config)# ip vrf vrf_blue
(config-vrf)# route-target both 1000:1
(config-vrf)# route-target export 1000:2
(config-vrf)# route-target import 173.27.0.130:200
```

Related Commands

Command	Description
ip vrf	Enters VRF configuration mode.
import	Configures an import route map for the VRF.

show ip bgp vpnv4

To display VPN address information from the BGP table, use the **show ip bgp vpnv4** EXEC command.

show ip bgp vpnv4 {**all** | **rd** *route-distinguisher* | **vrf** *vrf-name*}
[*ip-prefix/length* [**longer-prefixes**] [*output-modifiers*]]
[*network-address* [*mask*] [**longer-prefixes**] [*output-modifiers*]] [**cidr-only**] [**community**] [**community-list**] [**dampened-paths**] [**filter-list**] [**flap-statistics**] [**inconsistent-as**] [**neighbors**] [**paths** [*line*]] [**peer-group**] [**quote-regexp**] [**regexp**] [**summary**] [**tags**]

Syntax Description

all	Displays the complete VPNv4 database.
rd *route-distinguisher*	Displays NLRIs that have a matching route distinguisher.
vrf *vrf-name*	Displays NLRIs associated with the named VRF.
ip-prefix/length	(Optional) IP prefix address (in dotted decimal format) and length of mask (0 to 32).
longer-prefixes	(Optional) Displays the entry, if any, that exactly matches the specified prefix parameter, as well as all entries that match the prefix in a "longest-match" sense. That is, prefixes for which the specified prefix is an initial sub-string.
output-modifiers	(Optional) For a list of associated keywords and arguments, use context-sensitive help.
network-address	(Optional) IP address of a network in the BGP routing table.
mask	(Optional) Mask of the network address, in dotted decimal format.
cidr-only	(Optional) Displays only routes that have nonnatural net masks.
community	(Optional) Displays routes matching this community.
community-list	(Optional) Displays routes matching this community list.
dampened-paths	(Optional) Displays paths suppressed due to dampening (BGP route from peer is up and down).
filter-list	(Optional) Displays routes conforming to the filter list.
flap-statistics	(Optional) Displays flap statistics of routes.
inconsistent-as	(Optional) Displays only routes that have inconsistent autonomous systems of origin.
neighbors	(Optional) Displays details about TCP and BGP neighbor connections.
paths	(Optional) Displays path information.
line	(Optional) A regular expression to match the BGP AS paths.
peer-group	(Optional) Displays information about peer groups.
quote-regexp	(Optional) Displays routes matching the AS path "regular expression."

regexp	(Optional) Displays routes matching the AS path regular expression.
summary	(Optional) Displays BGP neighbor status.
tags	(Optional) Displays incoming and outgoing BGP labels for each NLRI.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Use this command to display VPNv4 information from the BGP database. The command **show ip bgp vpnv4 all** displays all available VPNv4 information. The command **show ip bgp vpnv4 summary** displays BGP neighbor status.

Examples

The following example shows output for all available VPNv4 information in a BGP routing table:

```
Router# show ip bgp vpnv4 all
BGP table version is 18, local router ID is 14.14.14.14
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP,? - incomplete

   Network          Next Hop     Metric LocPrf    Weight   Path
Route Distinguisher: 100:1 (vrf1)
*> 11.0.0.0         50.0.0.1       0                0      101 i
*>i12.0.0.0         13.13.13.13    0    100         0      102 i
*> 50.0.0.0         50.0.0.1       0                0      101 i
*>i51.0.0.0         13.13.13.13    0    100         0      102 i
```

Table 1 describes the fields shown in this example.

Table 1 Show IP BGP VPNv4 Field Descriptions

Field	Description
Network	Displays the network address from the BGP table.
Next Hop	Displays the address of the BGP next hop.
Metric	Displays the BGP metric.
LocPrf	Displays the local preference.
Weight	Displays the BGP weight.
Path	Displays the BGP path per route.

The following example shows how to display a table of labels for NLRIs that have a route-distinguisher value of 100:1.

```
Router# show ip bgp vpnv4 rd 100:1 tags
   Network          Next Hop      In tag/Out tag
Route Distinguisher: 100:1 (vrf1)
   2.0.0.0          10.20.0.60      34/notag
   10.0.0.0         10.20.0.60      35/notag
   12.0.0.0         10.20.0.60      26/notag
                    10.20.0.60      26/notag
   13.0.0.0         10.15.0.15      notag/26
```

Table 2 describes the fields shown in this example.

Table 2 Show IP BGP VPNv4 rd Tags Field Descriptions

Field	Description
Network	Displays the network address from the BGP table.
Next Hop	Specifies the BGP next hop address.
In Tag	Displays the label (if any) assigned by this router.
Out Tag	Displays the label assigned by the BGP next hop router.

The following example shows VPNv4 routing entries for the VRF called vrf1.

```
Router# show ip bgp vpnv4 vrf vrf1
BGP table version is 18, local router ID is 14.14.14.14
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP,? - incomplete

Network          Next Hop        Metric LocPrf    Weight    Path
Route Distinguisher: 100:1 (vrf1)
*> 11.0.0.0         50.0.0.1       0                0       101 i
*>i12.0.0.0         13.13.13.13    0    100         0       102 i
*> 50.0.0.0         50.0.0.1       0                0       101 i
*>i51.0.0.0         13.13.13.13    0    100         0       102 i
```

Table 3 describes the fields shown in this example.

Table 3 Show IP BGP VPNv4 Field Descriptions

Field	Description
Network	Displays network address from the BGP table.
Next Hop	Displays address of the BGP next hop.
Metric	Displays the BGP metric.
LocPrf	Displays the local preference.
Weight	Displays the BGP weight.
Path	Displays the BGP path per route.

Related Command

Command	Description
show ip vrf	Displays VRFs and associated interfaces.

show ip cef vrf

To display the CEF forwarding table associated with a VRF, use the **show ip cef vrf** EXEC command.

show ip cef vrf *vrf-name* [*ip-prefix* [*mask* [**longer-prefixes**]] [**detail**] [*output-modifiers*]] [*interface interface-number*] [**adjacency** [*interface interface-number*] [**detail**] [**discard**] [**drop**] [**glean**] [**null**] [**punt**] [*output-modifiers*]] [**detail** [*output-modifiers*]] [**non-recursive** [**detail**] [*output-modifiers*]] [**summary** [*output-modifiers*]] [**traffic** [**prefix-length**] [*output-modifiers*]] [**unresolved** [**detail**] [*output-modifiers*]]

Syntax Description

vrf-name	Name assigned to the VRF.
ip-prefix	(Optional) IP prefix of entries to show, in dotted decimal format (A.B.C.D).
mask	(Optional) Mask of the IP prefix, in dotted decimal format.
longer-prefixes	(Optional) Displays table entries for all of the more specific routes.
detail	(Optional) Displays detailed information for each CEF table entry.
output-modifiers	(Optional) For a list of associated keywords and arguments, use context-sensitive help.
interface	(Optional) Type of network interface to use: ATM, Ethernet, Loopback, POS (packet over SONET) or Null.
interface-number	Number identifying the network interface to use.
adjacency	(Optional) Displays all prefixes resolving through adjacency.
discard	Discards adjacency.
drop	Drops adjacency.
glean	Gleans adjacency.
null	Null adjacency.
punt	Punts adjacency.
non-recursive	(Optional) Displays only nonrecursive routes.
summary	(Optional) Displays a CEF table summary.
traffic	(Optional) Displays traffic statistics.
prefix-length	(Optional) Displays traffic statistics by prefix size.
unresolved	(Optional) Displays only unresolved routes.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Used with only the *vrf-name* argument, the **show ip cef vrf** command shows a shortened display of the CEF table.

Used with the **detail** argument, the **show ip cef vrf** command shows detailed information for all CEF table entries.

Example

This example shows the forwarding table associated with the VRF called vrf1.

```
Router# show ip cef vrf vrf1
Prefix              Next Hop            Interface
0.0.0.0/32          receive
11.0.0.0/8          50.0.0.1            Ethernet1/3
12.0.0.0/8          52.0.0.2            POS6/0
50.0.0.0/8          attached            Ethernet1/3
50.0.0.0/32         receive
50.0.0.1/32         50.0.0.1            Ethernet1/3
50.0.0.2/32         receive
50.255.255.255/32   receive
51.0.0.0/8          52.0.0.2            POS6/0
224.0.0.0/24        receive
255.255.255.255/32  receive
```

Table 4 describes the fields shown in this example.

Table 4 Show IP CEF vrf Field Descriptions

Field	Description
Prefix	Specifies the network prefix.
Next Hop	Specifies the BGP next hop address.
Interface	Specifies the VRF interface.

Related Commands

Command	Description
show ip route vrf	Displays the IP routing table associated with a VRF.
show ip vrf	Displays VRF interfaces.

show ip protocols vrf

To display the routing protocol information associated with a VRF, use the **show ip protocols vrf** EXEC command.

show ip protocols vrf *vrf-name*

Syntax Description

vrf-name	Name assigned to a VRF.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Use this command to display routing information associated with a VRF.

Example

The following example shows information about a VRF called vpn1:

```
Router# show ip protocols vrf vpn2
Routing Protocol is "bgp 100"
  Sending updates every 60 seconds, next due in 0 sec
  Outgoing update filter list for all interfaces is
  Incoming update filter list for all interfaces is
  IGP synchronization is disabled
  Automatic route summarization is disabled
  Redistributing:connected, static
  Routing for Networks:
  Routing Information Sources:
    Gateway         Distance      Last Update
    13.13.13.13          200      02:20:54
    18.18.18.18          200      03:26:15
  Distance:external 20 internal 200 local 200
```

Table 5 describes the fields shown in this example.

Table 5 Show IP Protocols vrf Field Descriptions

Field	Description
Gateway	Displays the IP address of the router identifier for all routers in the network.
Distance	Displays the metric used to access the destination route.
Last update	Displays the last time the routing table was updated from the source.

Related Command

Command	Description
show ip vrf	Displays VRF interfaces.

show ip route vrf

To display the IP routing table associated with a VRF (VPN routing/forwarding instance), use the **show ip route vrf** EXEC command.

show ip route vrf *vrf-name* [**connected**] [*protocol* [*as-number*] [*tag*] [*output-modifiers*]]
[**list** *number* [*output-modifiers*]] [**profile**] [**static** [*output-modifiers*]]
[**summary** [*output-modifiers*]] [**supernets-only** [*output-modifiers*]]
[**traffic-engineering** [*output-modifiers*]]

Syntax Description

vrf-name	Name assigned to the VRF.
connected	Displays all connected routes in a VRF.
protocol	To specify a routing protocol, use one of the following keywords: bgp, egp, eigrp, hello, igrp, isis, ospf, or rip.
as-number	Autonomous system number.
tag	IOS routing area label.
output-modifiers	(Optional) For a list of associated keywords and arguments, use context-sensitive help.
list *number*	Specifies the IP access list to display.
profile	Displays the IP routing table profile.
static	Displays static routes.
summary	Displays a summary of routes.
supernets-only	Displays supernet entries only.
traffic-engineering	Displays only traffic-engineered routes.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

This command displays specified information from the IP routing table of a VRF.

Examples

This example shows the IP routing table associated with the VRF called vrf1:

```
Router# show ip route vrf vrf1
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
       U - per-user static route, o - ODR
       T - traffic engineered route

Gateway of last resort is not set

B    51.0.0.0/8 [200/0] via 13.13.13.13, 00:24:19
C    50.0.0.0/8 is directly connected, Ethernet1/3
B    11.0.0.0/8 [20/0] via 50.0.0.1, 02:10:22
B    12.0.0.0/8 [200/0] via 13.13.13.13, 00:24:20
```

This example shows BGP entries in the IP routing table associated with the VRF called vrf1:

```
Router# show ip route vrf vrf1 bgp
B  51.0.0.0/8 [200/0] via 13.13.13.13, 03:44:14
B  11.0.0.0/8 [20/0] via 51.0.0.1, 03:44:12
B  12.0.0.0/8 [200/0] via 13.13.13.13, 03:43:14
```

Related Commands

Command	Description
show ip cef vrf	Displays the CEF forwarding table associated with a VRF.
show ip vrf	Displays VRFs and associated interfaces.

show ip vrf

To display the set of defined VRFs (VPN routing/forwarding instances) and associated interfaces, use the **show ip vrf** EXEC command.

show ip vrf [{**brief | detail** | **interfaces**}] [*vrf-name*] [*output-modifiers*]

Syntax Description

brief	(Optional) Displays concise information on the VRF(s) and associated interfaces.
detail	(Optional) Displays detailed information on the VRF(s) and associated interfaces.
interfaces	(Optional) Displays detailed information about all interfaces bound to a particular VRF, or any VRF.
vrf-name	Name assigned to a VRF.
output-modifiers	(Optional) For a list of associated keywords and arguments, use context-sensitive help.

Default

When no optional parameters are specified the command shows concise information about all configured VRFs.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Use this command to display information about VRFs. Two levels of detail are available: use the **brief** keyword or no keyword to display concise information, or use the **detail** keyword to display all information. To display information about all interfaces bound to a particular VRF, or to any VRF, use the **interfaces** keyword.

Examples

This example shows brief information for the VRFs currently configured:

```
Router# show ip vrf
  Name                Default RD          Interfaces
  vrf1                100:1               Ethernet1/3
  vrf2                100:2               Ethernet0/3
```

Table 6 describes the fields shown in this example.

Table 6 Show vrf Field Descriptions

Field	Description
Name	Specifies the VRF name.
Default RD	Specifies the default route distinguisher.
Interfaces	Specifies the network interfaces.

This example shows detailed information for the VRF called vrf1:

```
Router# show ip vrf detail vrf1
VRF vrf1; default RD 100:1
 Interfaces:
  Ethernet1/3
 Connected addresses are in global routing table
 Export VPN route-target communities
  RT:100:1
 Import VPN route-target communities
  RT:100:1
 No import route-map
```

Table 7 describes the fields shown in this example.

Table 7 Show IP vrf Detail Field Descriptions

Field	Description
Interfaces	Specifies the network interfaces.
Export	Specifies VPN route-target export communities.
Import	Specifies VPN route-target import communities.

This example shows the interfaces bound to a particular VRF:

```
router# show ip vrf interfaces
Interface       IP-Address      VRF                             Protocol
Ethernet2       130.22.0.33     blue_vrf                        up
Ethernet4       130.77.0.33     hub                             up
router#
```

Table 8 describes the fields shown in this example.

Table 8 Show IP vrf Interfaces Field Descriptions

Field	Description
Interface	Specifies the network interfaces for a VRF.
IP-Address	Specifies the IP address of a VRF interface.
VRF	Specifies the VRF name.
Protocol	Displays the state of the protocol (up/down) for each VRF interface.

Related Commands

Command	Description
ip vrf	Enters VRF configuration mode.
rd	Configures a default route distinguisher (RD) for a VRF.
route-target	Configures import and export extended community attributes for the VRF.
import	Configures an import route map for a VRF.
ip vrf forwarding	Associates a VRF with an interface or subinterface.

show tag-switching forwarding vrf

To display label forwarding information for advertised VRF routes, use the **show tag-switching forwarding vrf** EXEC command. To disable the display of label forwarding information, use the **no** form of this command.

show tag-switching forwarding vrf *vrf-name* [*ip-prefix/length* [*mask*]] [**detail**] [*output-modifiers*]

no show tag-switching forwarding vrf *vrf-name* [*ip-prefix/length* [*mask*]] [**detail**] [*output-modifiers*]

Syntax Description

vrf-name	Displays NLRIs associated with the named VRF.
ip-prefix/length	(Optional) IP prefix address (in dotted decimal format) and length of mask (0 to 32).
mask	(Optional) Destination network mask, in dotted decimal format.
detail	(Optional) Displays detailed information on the VRF routes.
output-modifiers	(Optional) For a list of associated keywords and arguments, use context-sensitive help.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Usage Guidelines

Use this command to display label forwarding entries associated with a particular VRF or IP prefix.

Example

The following example shows label forwarding entries that correspond to the VRF called vpn1:

```
Router# show tag-switching forwarding vrf vrf1 detail
```

Related Commands

Command	Description
show tag-switching forwarding	Displays label forwarding information.
show ip cef vrf	Displays VRFs and associated interfaces.

Debug Commands

This section documents new debug commands. All other commands used with this feature are documented in the Cisco IOS Release 12.0 command references.

- debug ip bgp

debug ip bgp

To display information related to processing BGPs, use the **debug ip bgp** EXEC command. To disable the display of BGP information, use the **no** form of this command.

debug ip bgp [*A.B.C.D.* | **dampening | events | in | keepalives | out | updates | vpnv4**]

no debug ip bgp [*A.B.C.D.* | **dampening | events | in | keepalives | out | updates | vpnv4**]

Syntax Description

A.B.C.D.	(Optional) Displays the BGP neighbor IP address.
dampening	(Optional) Displays BGP dampening.
events	(Optional) Displays BGP events.
in	(Optional) BGP inbound information.
keepalives	(Optional) Displays BGP keepalives.
out	(Optional) Displays BGP outbound information.
updates	(Optional) Displays BGP updates.
vpnv4	(Optional) Displays VPNv4 NLRI information.

Default

No default behavior or values.

Command Mode

EXEC

Command History

Release	Modification
12.0(5)T	This command was introduced.

Example

The following example displays the output from this command:

```
Router# debug ip bgp vpnv4
03:47:14:vpn:bgp_vpnv4_bnetinit:100:2:58.0.0.0/8
03:47:14:vpn:bnettable add:100:2:58.0.0.0 / 8
03:47:14:vpn:bestpath_hook route_tag_change for vpn2:58.0.0.0/255.0.0.0(ok)
03:47:14:vpn:bgp_vpnv4_bnetinit:100:2:57.0.0.0/8
03:47:14:vpn:bnettable add:100:2:57.0.0.0 / 8
03:47:14:vpn:bestpath_hook route_tag_change for vpn2:57.0.0.0/255.0.0.0(ok)
03:47:14:vpn:bgp_vpnv4_bnetinit:100:2:14.0.0.0/8
03:47:14:vpn:bnettable add:100:2:14.0.0.0 / 8
03:47:14:vpn:bestpath_hook route_tag_chacle ip bgp *nge for vpn2:14.0.0.0/255.0.0.0(ok)
```

Glossary

ATM-LSR—A label switch router with a number of LSC-ATM interfaces. The router forwards the cells among these interfaces using labels carried in the VPI/VCI field.

ATM edge LSR—A router that is connected to the ATM-LSR cloud through LSC-ATM interfaces. The ATM edge LSR adds labels to unlabeled packets and strips labels from labeled packets.

BGP—Border Gateway Protocol. Interdomain routing protocol that exchanges reachability information with other BGP systems. It is defined in RFC 1163.

CEF—Cisco Express Forwarding. An advanced Layer 3 IP switching technology. CEF optimizes network performance and scalability for networks with large and dynamic traffic patterns.

CE router—Customer edge router. A router that is part of a customer network and that interfaces to a provider edge (PE) router. CE routers are not aware of associated VPNs.

CoS—Class of Service. A feature that provides scalable, differentiated types of service across an MPLS network.

GRE—Generic routing encapsulation. A tunneling protocol developed by Cisco that can encapsulate a wide variety of protocol packet types inside IP tunnels, creating a virtual point-to-point link to Cisco routers at remote points over an IP internetwork. By connecting multiprotocol subnetworks in a single-protocol backbone environment, IP tunneling that uses GRE allows network expansion across a single-protocol backbone environment.

IGP—Interior Gateway Protocol. An Internet protocol used to exchange routing information within an autonomous system. Examples of common IBGPs include IGRP, OSPF, and RIP.

IS-IS—Intermediate system-to-intermediate system. OSI link-state hierarchical routing protocol in which ISs (routers) exchange routing information based on a single metric to determine network topology.

Label-switched path (LSP)—A sequence of hops (R0...Rn) in which a packet travels from R0 to Rn through label switching mechanisms. A label-switched path can be established dynamically, based on normal routing mechanisms, or through configuration.

Label-switched path (LSP) tunnel—A configured connection between two routers, in which MPLS is used to carry the packet.

LSA—Link-state advertisement. A broadcast packet used by link-state protocols. The LSA contains information about neighbors and path costs and is used by the receiving router to maintain a routing table.

MPLS—Multiprotocol Label Switching. An emerging industry standard.

NLRI—Network layer reachability information. BGP sends routing update messages containing NLRI to describe a route and how to get there. In this context, an NLRI is a prefix. A BGP update message carries one or more NLRI prefixes and the attributes of a route for the NLRI prefixes; the route attributes include a BGP next hop gateway address, community values, and other information.

PE router—Provider edge router. A router that is part of a service provider's network connected to a customer edge (CE) router. All VPN processing occurs in the PE router.

RD—Route distinguisher. An 8-byte value that is concatenated with an IPv4 prefix to create a unique VPN IPv4 prefix.

RIP—Routing Information Protocol. An IGP used to exchange routing information within an autonomous system, RIP uses hop count as a routing metric.

traffic engineering—The techniques and processes used to cause routed traffic to travel through the network on a path other than the one that would have been chosen if standard routing methods had been used.

traffic engineering tunnel—A label-switched path tunnel that is used for engineering traffic. It is set up through means other than normal Layer 3 routing and is used to direct traffic over a path different from the one that Layer 3 routing would cause it to take.

tunneling—Architecture providing the services necessary to implement any standard point-to-point data encapsulation scheme.

VPN—Virtual private network. A secure IP-based network that shares resources on one or more physical networks. A VPN contains geographically dispersed sites that can communicate securely over a shared backbone.

vpnv4—Used as a keyword in commands to indicate VPN-IPv4 prefixes. These prefixes are customer VPN addresses, each of which has been made unique by the addition of an 8-byte route distinguisher.

VRF—VPN routing/forwarding instance. A VRF consists of an IP routing table, a derived forwarding table, a set of interfaces that use the forwarding table, and a set of rules and routing protocols that determine what goes into the forwarding table. In general, a VRF includes the routing information that defines a customer VPN site that is attached to a PE router.